Web page basics

# Lesson Notes

## How a web browser works

* We are going to be creating web pages in this class. We will start out by just learning how to format a page, and then we will work on adding a programs to our pages. In order to do this, we need to understand how a web browser works.
* When you navigate in your web browser, your web browser will ask some computer on the internet for something to display. That something is called a **resource**. A resource is a generic term for any type of data: a web page, an image, a sound, etc.
* Web browsers have a standard way of referencing a resource called a **URL**, which stands for Uniform Resource Locator. A URL is made up of several parts.
* The first part of a URL identifies how the web browser should communicate with the computer on the internet to get the resource. Just like when you meet someone you put out your hand to shake their hand, there are a set of rules for how the web browser should communicate. We call these rules a **protocol**.
* One common protocol is HTTP, which stands for Hypertext Transfer Protocol.
* The next part of the URL is called the **hostname**, which is used to identify which computer to communicate with. Depending on the web site, there may be more than one computer that is able to answer the request from your browser. Each of the computers will respond the same basic way, so it doesn’t matter which one your browser communicates with. For example, large web sites may have thousands or even tens of thousands of computers.
* The next part of the URL is the path, which says which resource we are interested in for this particular site. A path of “/” means we want the home page.
* When you navigate to <http://universityprep.org/>, your web browser will first use the HTTP protocol to ask the computer at universityprep.org for the home page (/). The universityprep.org computer will answer with a resource that describes how the web page should look. The web browser will then use that description to display the web page. There are many ways of describing how a page should look. A common way used on the web is Hypertext Markup Language, or **HTML**.
* In order to describe the entire universityprep.org home page, it takes quite a bit of HTML. I’m just showing part of the HTML, with the rest shown as “…”.
* You will notice the HTML returned from universityprep.org contains text like: “Today is a NO SCHOOL DAY”. The browser can display this text.
* But how does it know where? There are other parts of HTML that give it this information, called **markup**. Markup typically appears between brackets (< and >) in the HTML.

## A closer look at HTML

* In order to create a web page, we have to learn how to write HTML. HTML is a language with a set of rules for how it is written, called a grammar. We have to follow that grammar when writing HTML, or the web browser may not understand what we want displayed. A web page written in HTML is called an **HTML document**.
* Let’s look at the grammar of a simple HTML document.
* The first part (<!DOCTYPE html>) identifies the version of the HTML document. We are using HTML5, but there have been other versions. Having a version in HTML is important, since it allows each web site on the internet to choose what version they want to use. This is much more practical than having to have all web sites switch at once.
* What follows the DOCTYPE is an HTML **element**, which represents the whole HTML document. HTML has a very regular structure consisting of elements and text. The key thing to notice that except for the HTML element, all the other elements are **contained** within some element. For example, the body element is contained within the html element. A paragraph element (abbreviated “p”) is contained within the body element. An image element (abbreviated “img”) is contained within the paragraph. This concept of containment has a big effect on how things display.
* The first thing you may notice is that each of elements contained within the body are displayed in the order in which they appear in the document. First comes the heading 1, then the first image, then the first paragraph, followed by the second paragraph. Each element **flows** one after another.
* As an example, if you made the image smaller, the rest of the page would automatically reflow. The ability for HTML to automatically flow pages is an important feature in today’s world of many screen sizes.
* Notice the difference between the two images. The first image element is contained within the body, and the second image is contained within a paragraph. The second image is drawn along with the text of the paragraph since it is contained by the paragraph.
* Let’s take a closer look at the syntax for an HTML file. Element that can contain other elements (for example, the html element), have both a start tag and an end tag. Elements that are not containers have only a start tag (for example the image tag).
* Most element types have additional values you can add to the start tag to change how they behave. These values are called **attributes.** For example, the “img” tag has a “src” attribute which specifies the location of the image to display.
* Next we are going to do an exercise which involve creating HTML documents in a folder on your local computer. A web browser can either display a web site from the internet, or it can display a web site that is made up of files in a folder on your laptop. We are going to get started by just creating a web site in a folder since it is fast and easy to work with. Later in the class we will be creating real web sites on the internet.